Claim 4. A system for protecting memory from being written as claimed in Claim 1 in which the hardware means comprises:

a look-aside buffer including a plurality of storage locations for virtual addresses and associated physical addresses, and

a storage position in each storage location of the translation look aside buffer; and

in which the software means for protecting against writing the memory address invalidates translations associated with the memory address

Claim 5. A system for protecting memory from being written as claimed in Claim 1 in which the software means for protecting against writing the memory address removes translations associated with the memory address.

Claim 6. A system for protecting memory from being written as claimed in Claim 1 in which the hardware means comprises:

a look-aside buffer including a plurality of storage locations for virtual addresses and associated physical addresses, and

a storage position in each storage location of the translation look aside buffer; and

in which the software means for protecting against writing the memory address removes translations associated with the memory address.

Claim 7. A computer system comprising:

a host processor designed to execute instructions of a host instruction set,

software for translating instructions from a target instruction set to instructions of the host instruction set,

memory for storing target instructions from a program being translated,

a translation buffer for storing host instructions translated from target instructions for execution, and

hardware means for generating an exception to a write access to a target address storing a target instruction which has been translated to a host instruction.

Claim 7 8. A computer system as claimed in Claim 6 7 in which the hardware means for generating an exception comprises a translation look-aside buffer including a plurality of storage locations for virtual and physical addresses of recently accessed memory, each of the storage location including a storage position for indicating that an instruction at a target address has been translated to a host instruction.

Claim 8 9. A computer system as claimed in Claim 6 7 further comprising software means responding to an exception to a write access to a target address storing a target instruction which has been translated to a host instruction for protecting against writing the memory address until it has been assured that translations associated with the memory address will not be utilized before being updated.

Claim 10. A computer system as claimed in Claim 9 in which the software means responding to an exception to a write access comprises software means invalidating translations associated with the memory address.

Claim 11. A computer system as claimed in Claim 9 in which the software means responding to an exception to a write access comprises software means removing translations associated with the memory address.

Claim 9/12. A method of responding to an attempt to write a memory address including a target instruction which has been translated to a host instruction for execution by a host processor including the steps of:

marking a memory address including a target instruction which has been translated to a host instruction,

detecting a memory address which has been marked when an attempt is made to write to the memory address, and

responding to the detection of a memory address which has been marked by protecting a target instruction at the memory address until it has been assured that translations associated with the memory address will not be utilized before being updated .

Claim 10/13. A method as claimed in Claim 9/12 in which the step of marking a memory address including a target instruction which has been translated to a host instruction comprises storing an indication that a target address has been translated in a memory location of a

translation look-aside buffer with a physical address of the target instruction.

Claim 14. A method as claimed in Claim 12 in which the step of responding to the detection of a memory address which has been marked by protecting a target instruction at the memory address until it has been assured that translations associated with the memory address will not be utilized before being updated comprises the steps of:

generating an exception in response to the detection of a memory address which has been marked, and

responding to the exception by invalidating translations associated with the memory address before writing the memory address.

Claim 15. A microprocessor comprising:

a host processor capable of executing a first instruction set,

code morphing software for translating programs written for a target processor having a second different instruction set into instructions of the first instruction set for execution by the host processor, and

a memory controller comprising

an address translation buffer including a plurality of storage locations in which recently accessed virtual target addresses and physical memory addresses represented by the virtual target addresses are to be recorded,

Docket No. TM-95-050

Claim 11. A memory controller comprising

an address translation buffer including a plurality of storage locations in which recently accessed virtual addresses and physical addresses represented by the virtual addresses are to be recorded,

each of the storage locations including means for indicating whether a physical address stores an instruction of a target instruction set which has been translated to an instruction of a host instruction set; and

means for detecting an indication in a storage location to prevent a write access of to the physical address and for indicating a subsequent operation ~~beforeaccessing~~ before accessing the address.

Claim 19. A memory controller as claimed in Claim 18 in which the means for detecting an indication in a storage location to prevent a write access of to the physical address and for indicating a subsequent operation before accessing the address comprises

means for generating an exception in response to detection of an indication, and

means for responding to the exception to indicate a subsequent operation to be taken with respect to the translated host instruction before accessing the address.

Claim 20. A memory controller as claimed in Claim 11 in which the means for indicating comprises a storage position in a storage location.

Claim 1. A system for protecting memory from being written in a computer which includes a host processor designed to execute instructions of a host instruction set and software for translating instructions from a target instruction set to instructions of the host instruction set comprising:

hardware means for indicating whether a memory address stores a target instruction which has been translated to host instructions, and

software means responding to an indication that a memory address stores a target instruction which has been translated to host instructions for protecting against writing the memory address until it has been assured that translations associated with the memory address will not be utilized before being updated once the memory address has been written.

Claim 2. A system for protecting memory from being written as claimed in Claim 1 in which the hardware means comprises:

a look-aside buffer including a plurality of storage locations for virtual addresses and associated physical addresses, and

a storage position in each storage location of the translation look aside buffer.

Claim 3. A system for protecting memory from being written as claimed in Claim 1 in which the software means for protecting against writing the memory address invalidates translations associated with the memory address.

each of the storage locations including means for indicating whether a target instruction at a physical address has been translated to a host instruction; and

means for responding to a write access of an address in a storage location of the address translation buffer in which the means for indicating indicates that a target instruction at a physical address has been translated to a host instruction for protecting against writing the memory address until it has been assured that translations associated with the memory address will not be utilized before being updated.

Claim 16. A microprocessor as claimed in Claim 15 in which the means for responding to a write access of an address in a storage location of the address translation buffer in which the means for indicating indicates that a target instruction at a physical address has been translated to a host instruction for protecting against writing the memory address comprises

means for generating an exception in response to a detection of an indication, and

means for responding to the exception to indicate a subsequent operation to be taken before accessing the memory address.

Claim 17. A microprocessor as claimed in Claim 16 in which the means for indicating whether a target instruction at a physical address has been translated to a host instruction comprises a storage position in a storage location.